1. レポートに意味づけをして下さい。

準備

課題１　HTML\_自習課題１を作成

SQLファイル名：自分の出席番号\_復習1.sql

課題番号、出席番号と氏名を1行目に表記

## 

* 1. employees表のデータを全て表示。(レイアウトを整えること)
  2. employees表の「employee\_id」「last\_name」「salary」列のデータを全て表示。
  3. employees表の「department\_id」列のデータを全て表示。ただし、重複データなしとする。
  4. employees表から職種（job\_id）が'SA\_REP'ではない社員の「employee\_id」「last\_name」「job\_id」列を表示。
  5. employees表から「last\_name」列および「salary」列と「commission\_pct」列を足した値を全て表示。そのときの列見出しは「社員名」「支給額」とすること。
  6. employees表から「hire\_date」が'98-04-01'から'99-03-31'の社員の「employee\_id」「last\_name」「hire\_date」列を表示。
  7. employees表から「department\_id」が10か20の社員の全ての列を表示。INを使用すること。
  8. employees表から「last\_name」がsで終わる社員の全ての列を表示。
  9. employees表から「manager\_id」がNULLの社員の「employee\_id」「last\_name」「job\_id」列を表示。
  10. employees表から「manager\_id」がNULL以外の社員の「employee\_id」「last\_name」「job\_id」列を表示。
  11. employees表から「salary」が5000～10000以外の社員の全ての列を表示。
  12. employees表から「last\_name」「manager\_id」「department\_id」列の全てを「department\_id」の昇順、さらに「manager\_id」の降順に表示。
  13. employees表から全社員の「last\_name」「salary」「salary + commission\_pct」列を表示する。ただし、「salary + commission\_pct」列が空白表示にならないようにすること。

COALESCE関数を使って「null」を「0」に変換

※NVL関数を使ってもいいが、NVL関数はOracle独自関数なので実務では使わないほうがいい。

* 1. employees表から「job\_id」が'ST\_CLERK'の社員の「last\_name」「salary」「salary/22」（小数点以下2桁切捨て）列を表示。

TRUNC関数で切捨て

* 1. 本日の日付、曜日、時間を表示。表示形式は日付が'yy/mm/dd'、曜日が'day'、時間が'hh24:mi:ss'とすること。

TO\_CHAR関数で型変換

* 1. employees表から「department\_id」が20である社員の「last\_name」「hire\_date」列と列名が「入社３年後」の列を表示。「入社３年後」の列には「hire\_date」から３年（36ヶ月）経った日付を表示すること。

ADD\_MONTHS関数

「入社3年後」は列別名

* 1. 今月の最後の日付を表示。

LAST\_DAY(SYSDATE) テーブルは「dual」表を使う

* 1. employees表から全員の「last\_name」「salary」列を連結し１つの列として表示。なお、「last\_name」の右には 'さんの給与は、' 、「salary」の右には 'ドルです。' と表示すること。
  2. departments表から全部門の「department\_name」列を表示。「department\_name」は、最初の文字を大文字にし、残りはすべて小文字で表示すること。

INITCAP関数

* 1. employees表から「job\_id」が'\_MAN'で終わる社員の「employee\_id」「last\_name」列を表示。「last\_name」は先頭から３文字の文字列を抜き出して表示すること。

SUBSTR関数

* 1. employees表から「last\_name」「job\_id」「管理職」を表示。管理職は、job\_id列の最後４文字が「PRES」であれば'社長'、「\_MAN」であれば'マネージャー'、それ以外なら'平社員'と表示すること。

※この問題はCASE式を使うこと。

* 1. employees表から社長（manager\_idがNULL）以外である社員の「last\_name」「job\_id」「salary」「給与ランク」を、給与の昇順で表示。給与ランクはsalaryが6000未満なら'安月給'、6000以上11000以下なら'普通'、11000より多ければ'高給取り'と表示すること。
  2. 設問２３のSELECT文を、salaryの100の位を四捨五入した値で給与ランクを付けるよう修正。

ROUND関数